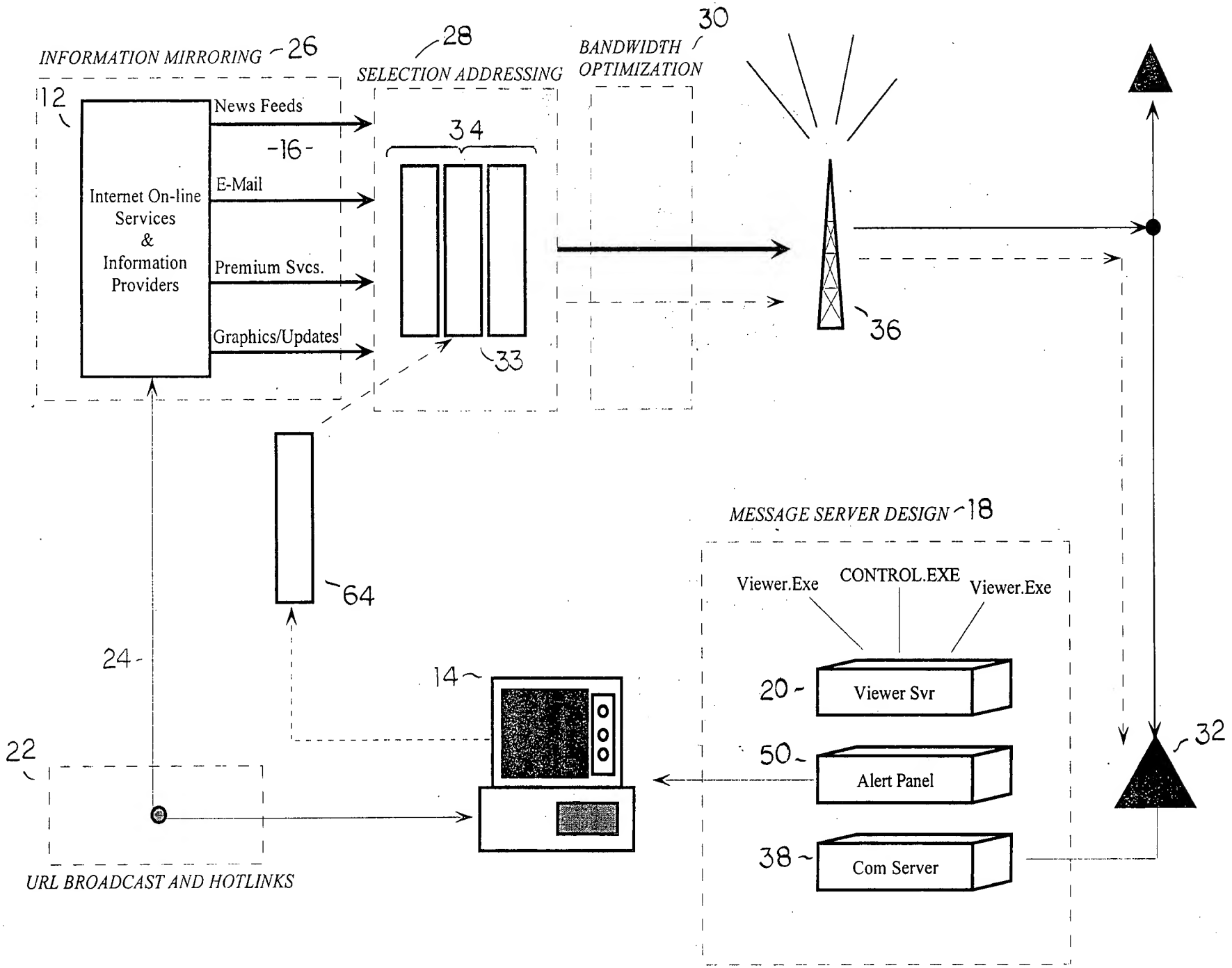

INFORMATION MIRRORING
26
SELECTION ADDRESSING
28
BANDWIDTH OPTIMIZATION
30
12
News Feeds
-16-
E-Mail
Premium Svcs.
Graphics/Updates
Internet On-line Services & Information Providers
34
33
36
64
MESSAGE SERVER DESIGN
18
Viewer.Exe
CONTROL.EXE
Viewer.Exe
20
Viewer Svr
50
Alert Panel
38
Com Server
32
24
14
22
URL BROADCAST AND HOTLINKS

FIG. 1

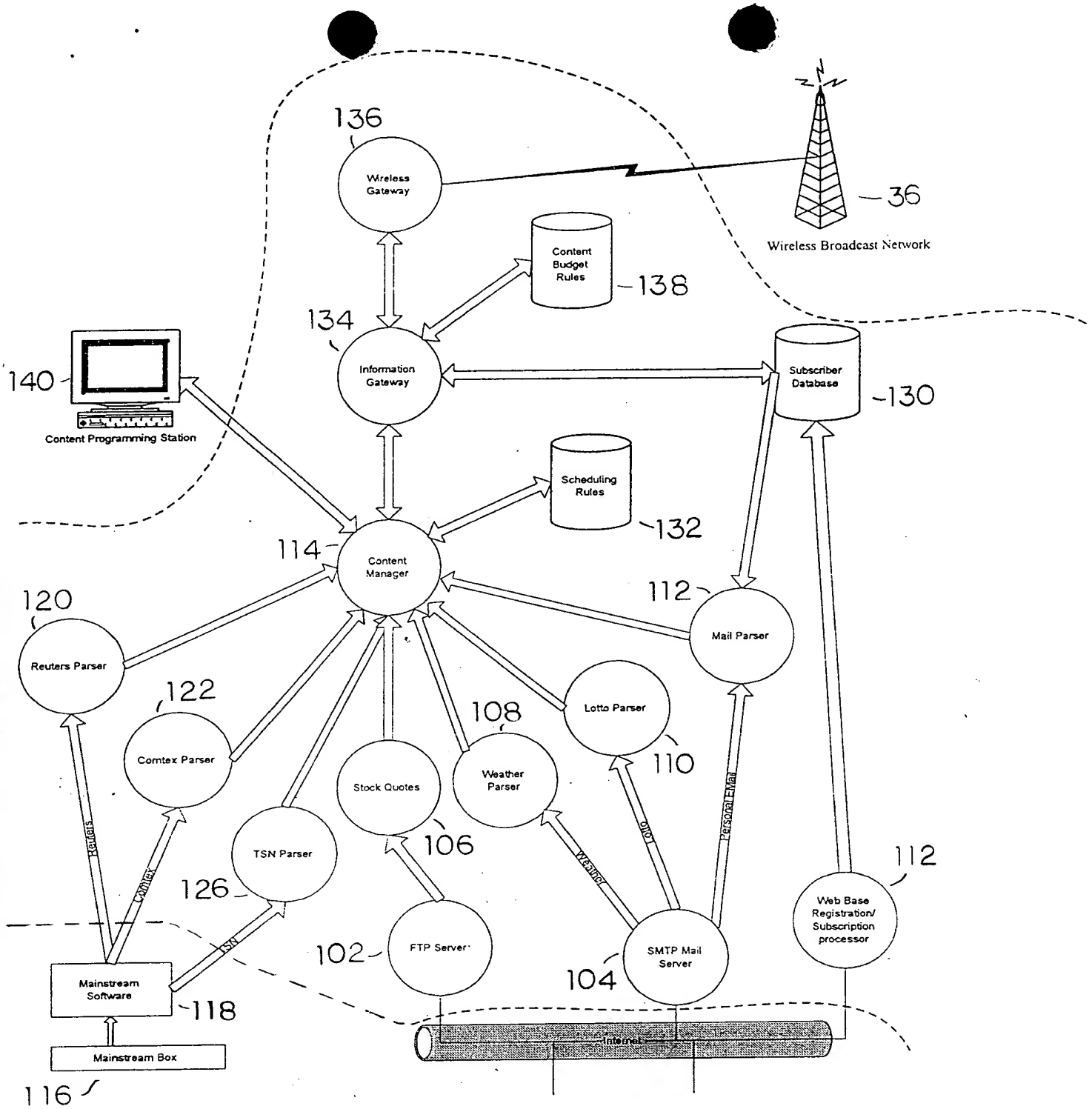

136
Wireless Gateway
36
Wireless Broadcast Network
Content Budget Rules
138
134
Information Gateway
140
Content Programming Station
Subscriber Database
130
Scheduling Rules
132
114
Content Manager
112
120
Reuters Parser
Mail Parser
122
Comtex Parser
108
Lotto Parser
110
Stock Quotes
Weather Parser
106
TSN Parser
126
Reuters
Comtex
TSN
Weather
Lotto
PersonalEMail
112
Web Base Registration/ Subscription processor
102
FTP Server
SMTP Mail Server
104
Mainstream Software
118
Mainstream Box
116
Internet
12
100

FIG. 2

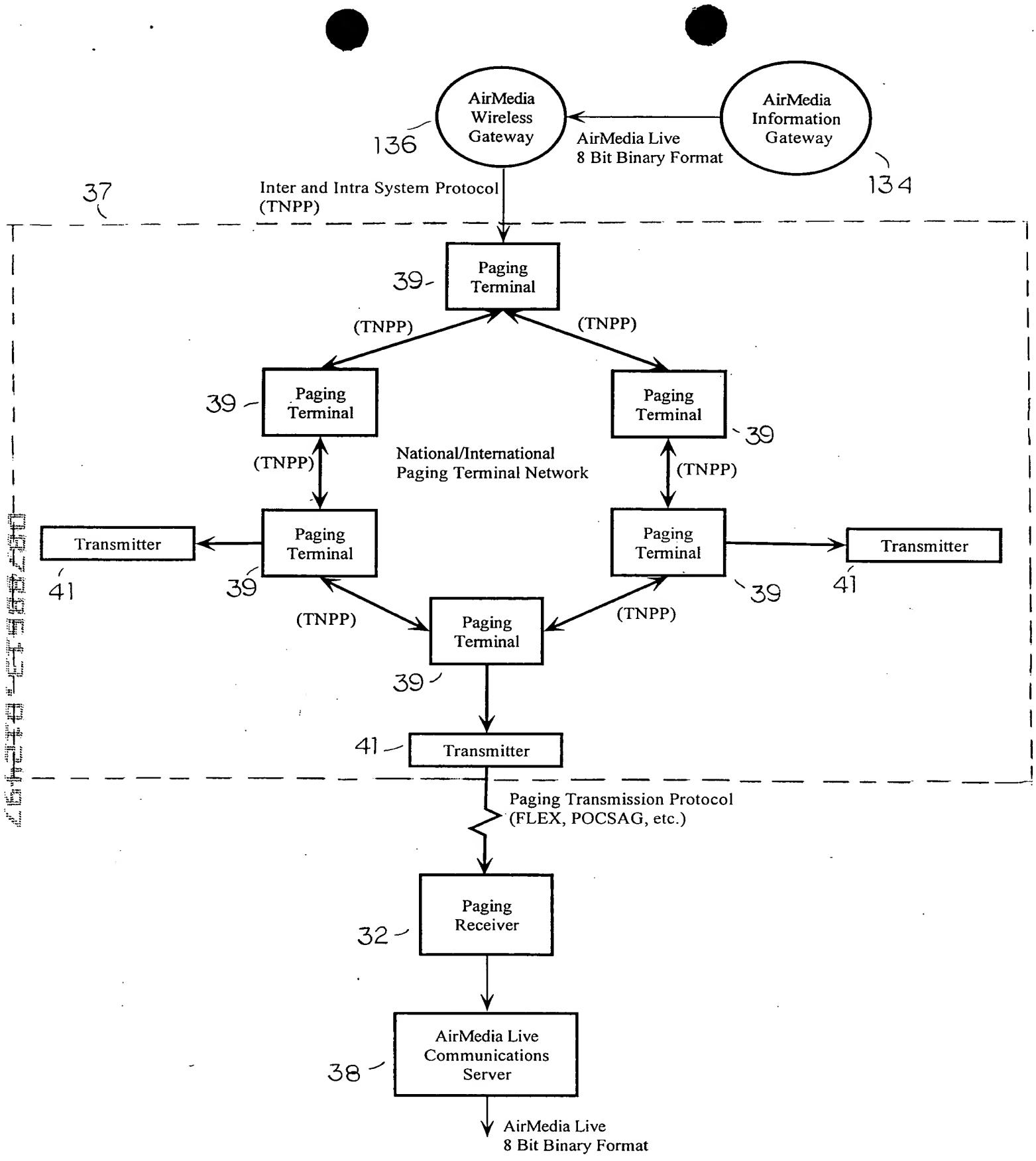

AirMedia Wireless Gateway
136
AirMedia Information Gateway
134
AirMedia Live 8 Bit Binary Format
37
Inter and Intra System Protocol (TNPP)
Paging Terminal
39
(TNPP)
(TNPP)
Paging Terminal
39
Paging Terminal
39
(TNPP)
National/International Paging Terminal Network
(TNPP)
Transmitter
41
Paging Terminal
39
Paging Terminal
39
Transmitter
41
(TNPP)
(TNPP)
Paging Terminal
39
Transmitter
41
Paging Transmission Protocol (FLEX, POCSAG, etc.)
Paging Receiver
32
AirMedia Live Communications Server
38
AirMedia Live 8 Bit Binary Format

FIG. 3(a)

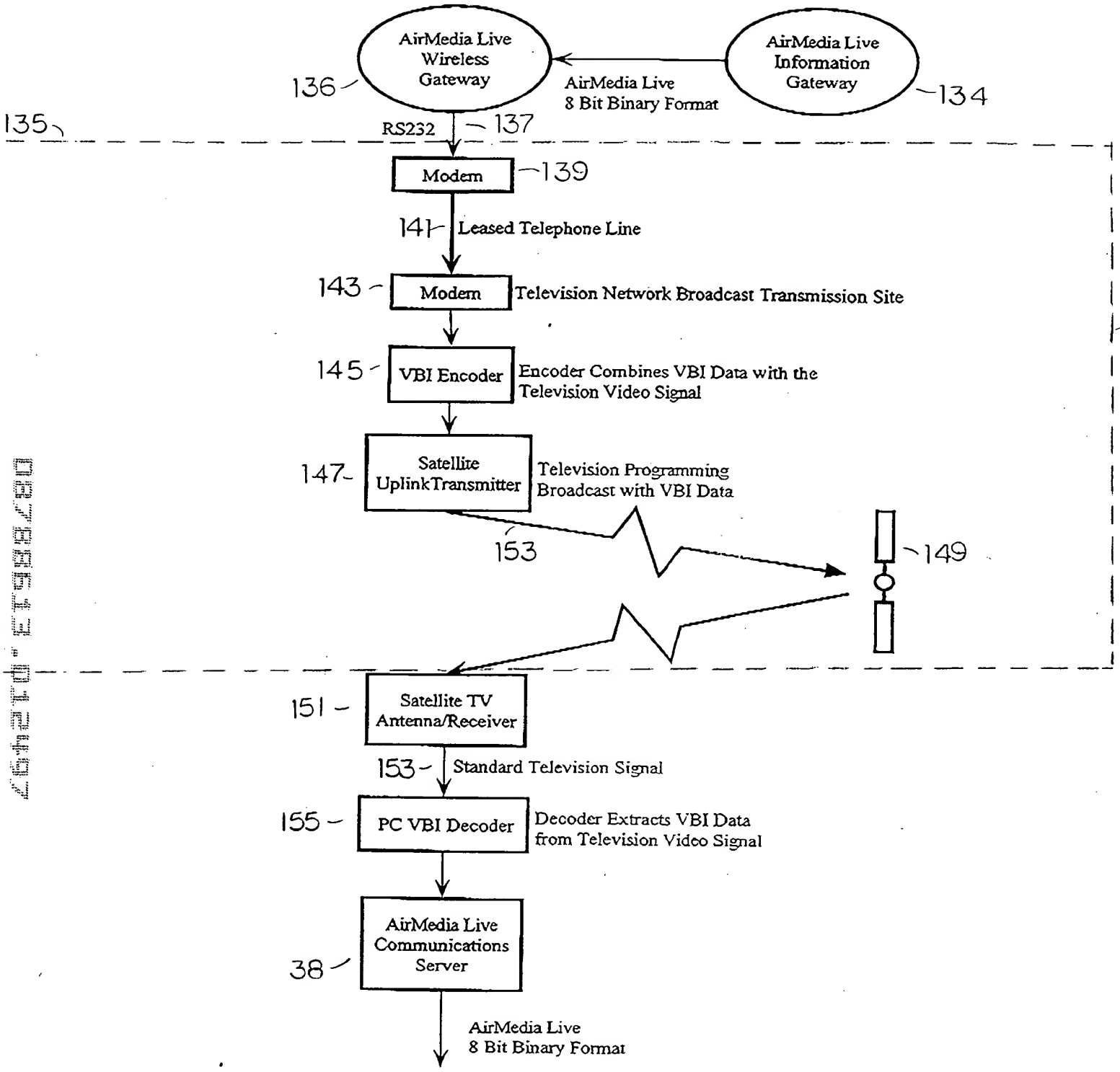

AirMedia Live Wireless Gateway
136
AirMedia Live Information Gateway
134
AirMedia Live 8 Bit Binary Format
135
RS232
137
Modem
139
141
Leased Telephone Line
143
Modem
Television Network Broadcast Transmission Site
145
VBI Encoder
Encoder Combines VBI Data with the Television Video Signal
147
Satellite UplinkTransmitter
Television Programming Broadcast with VBI Data
153
149
151
Satellite TV Antenna/Receiver
153
Standard Television Signal
155
PC VBI Decoder
Decoder Extracts VBI Data from Television Video Signal
38
AirMedia Live Communications Server
AirMedia Live 8 Bit Binary Format

FIG. 3(b)

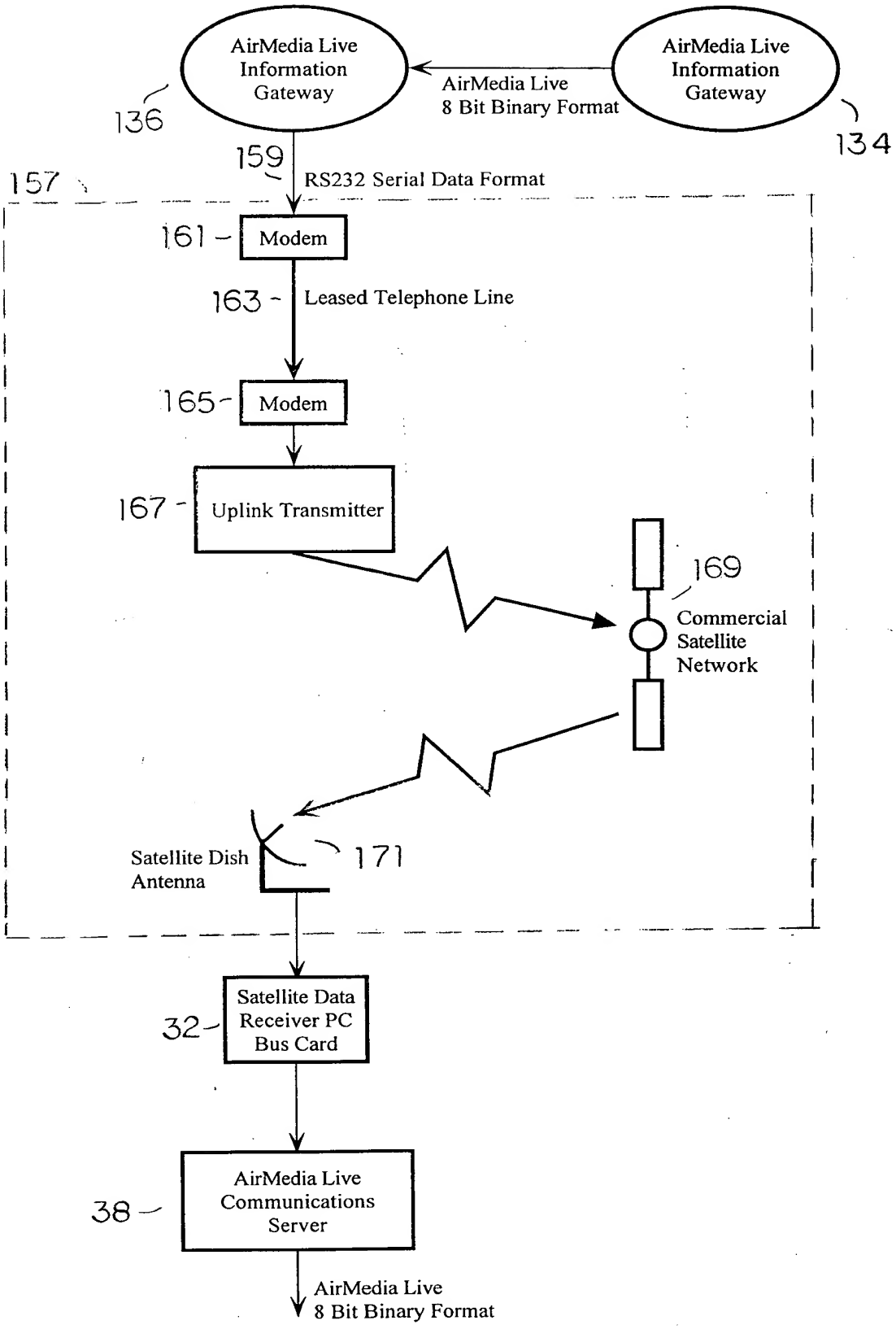


FIG 3(c)

Doc#: 2004283497

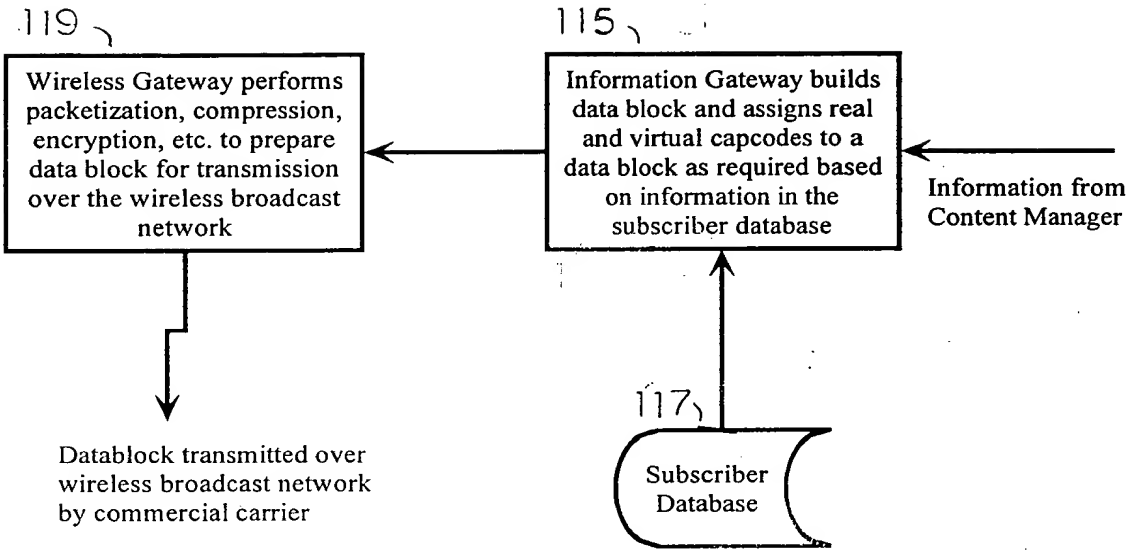

119
Wireless Gateway performs packetization, compression, encryption, etc. to prepare data block for transmission over the wireless broadcast network
115
Information Gateway builds data block and assigns real and virtual capcodes to a data block as required based on information in the subscriber database
Information from Content Manager
Datablock transmitted over wireless broadcast network by commercial carrier
117
Subscriber Database

FIG. 4

Item	Size	Description
Header:		
CRC	2 bytes	Standard Cyclical Redundancy Code to verify data block integrity.
Header Type	1 bit	If bit clear, then this is a message header. If bit set, then this is the data block header.
Custom Header Flag	1 bit	If bit clear, no custom header. If bit set, then a custom header is included in the data block.
Version Number	4 bits	Protocol version used.
Private Data Block Flag	1 bit	If bit clear, then this data block will be passed on to the Alert Panel for processing and display. If bit set, then this is a private data block to be processed internally by the Communications Server.
Virtual Capcode Flag	1 bit	If bit clear, then this data block is not targeted for a specific virtual capcode and no virtual capcode is included in the data block. If bit set, then this data block contains a virtual capcode.
Data Block Type	1 byte	The value of this byte specifies the type of data contained in the data block. If Private Data Block Flag is clear: 1 = plain text, 2 = AirMedia Live data feed format. If Private Data Block Flag is set: 1 = Capcode reprogramming message, 2 = Binary file transfer.
Data Block Version	4 bits	The version number of this data block's format.
Use Compression Flag	1 bit	If bit clear, then this data block is not compressed. If bit set, then compression is used and the compression type is specified in the Compression ID item.
Use Encryption Flag	1 bit	If bit clear, then this data block is not encrypted. If bit set, then this data block is encrypted.
Spare	2 bits	Reserved for future use.
Compression ID (optional)	1 byte	Included only if Use Compression Flag is set. Indicates the type of compression used.
Virtual Capcode (optional)	1 byte	Included only if Virtual Capcode flag set. Contains the virtual capcode to which this data block is targeted.
Size of Custom Header (optional)	1 byte	Included only if Custom Header Flag is set. Contains the size in bytes of the custom header.
Custom Header (optional)	variable	Reserved for future enhancements to data block protocol. Size determined from previous item.
Contents:		
Data Block Contents	variable	Information notification data from the information source to be processed by AirMedia Live software.

D 6,396,413 B2

FIG. 5

Item	Size	Description
Header:		
Alert Length	1 byte	The size of the alert data in bytes.
Alert Type	1 byte	The value of this item defines the alert type (e.g. new email arrival alert). Up to 256 predefined alert types are allowed.
Contents:		
Alert Data	variable	Personal alert notification data. Size of data is determined by the Alert Length item.

FIG. 6

09/754,260 - 05/27/2005

Item	Size	Description
Header:		
CRC	2 bytes	Standard Cyclical Redundancy Code to verify message integrity.
Header Type	1 bit	If bit clear, then this is a message header. If bit set, then this is the data block header.
Custom Header Flag	1 bit	If bit clear, no custom header. If bit set, then a custom header is included in the message.
Data Block ID	14 bits	ID of the data block to which this message belongs.
Message Number	1 byte	The position of this message in the data block (i.e. message sequence number).
Total Messages	1 byte	Total number of messages in the data block.
Size of Custom Header (optional)	1 byte	Included only if Custom Header Flag is set. Contains the size in bytes of the custom header.
Custom Header (optional)	variable	Reserved for future enhancements to message protocol. Size determined from previous item.
Contents:		
Message Contents	variable	The data portion of the message.

FIG. 7

Item	Size	Description
Header:		
Packet Type	4 bits	The value of this item indicates the packet type: 0 = Standard AirMedia Live Packet; 1 = Single Packet Data Block; if the left most bit (high bit) is set, then this is a Binary Alert Packet.
Total Packets Flag	1 bit	If bit is clear, then the Total Data Packets and Total Error Correction Packets items are not present. If bit is set, then the Total Data Packets and Total Error Correction Packets items are present.
Message ID	11 bits	The number of the message to which this packet belongs.
Packet Number	1 byte	The position of this packet in the message (packet sequence number).
Total Data Packets	1 byte	Total number of data packets in the message (does not include error correction packets).
Total Error Correction Packets	1 byte	Total number of Reed-Solomon forward error correction packets in the message.
Contents:		
Packet Contents	variable	The data portion of the packet.

FIG. 8

Item	Size	Description
Header:		
Packet Type	4 bits	The value of this item indicates the packet type: 0 = Standard AirMedia Live Packet; 1 = Single Packet Data Block; if the left most bit (high bit) is set, then this is a Binary Alert Packet.
Data Block ID	12 bits	The ID of the data block contained in this packet.
Contents:		
Packet Contents	variable	The header and contents of the data block contained in this packet.

FIG. 9

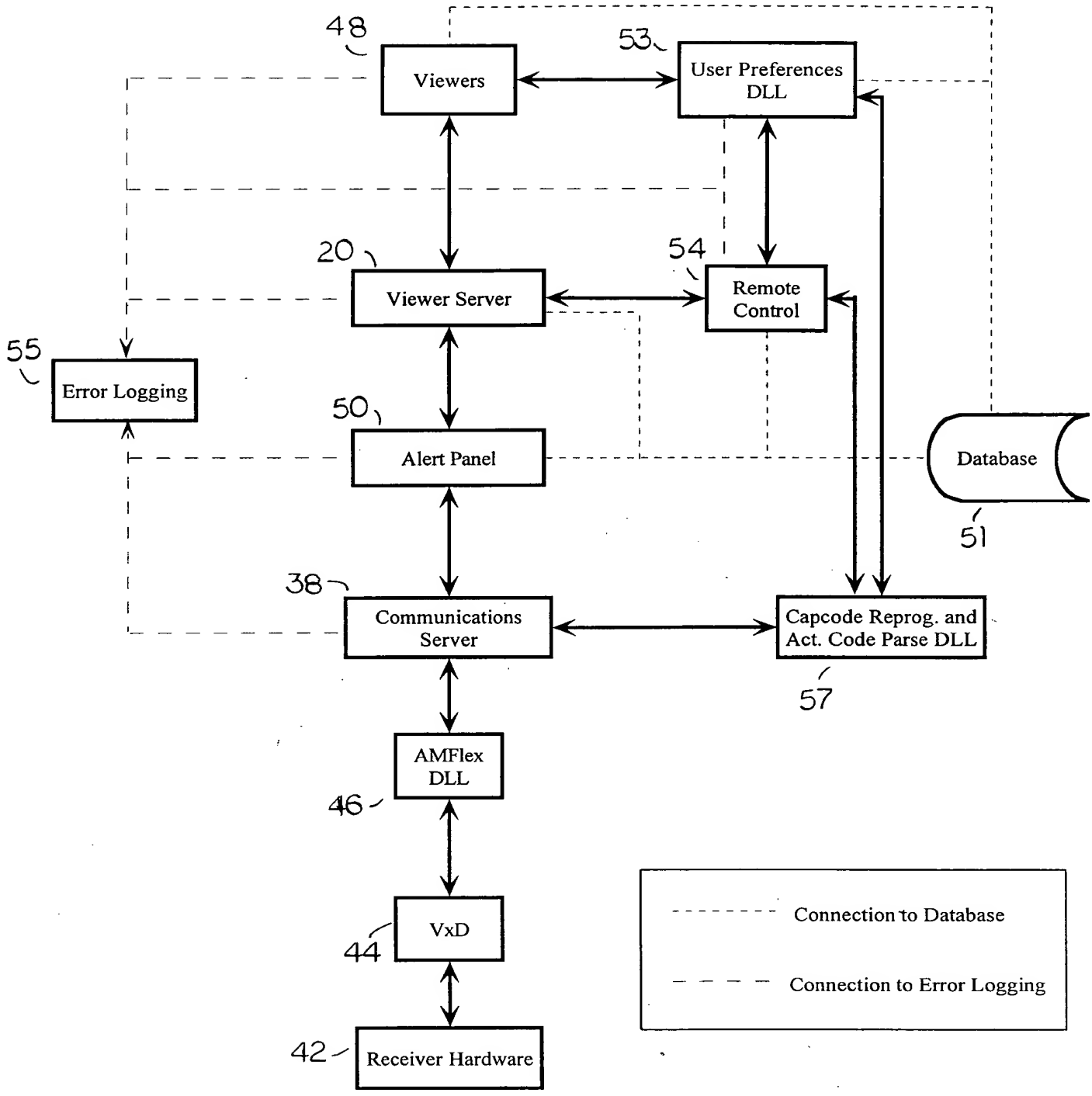
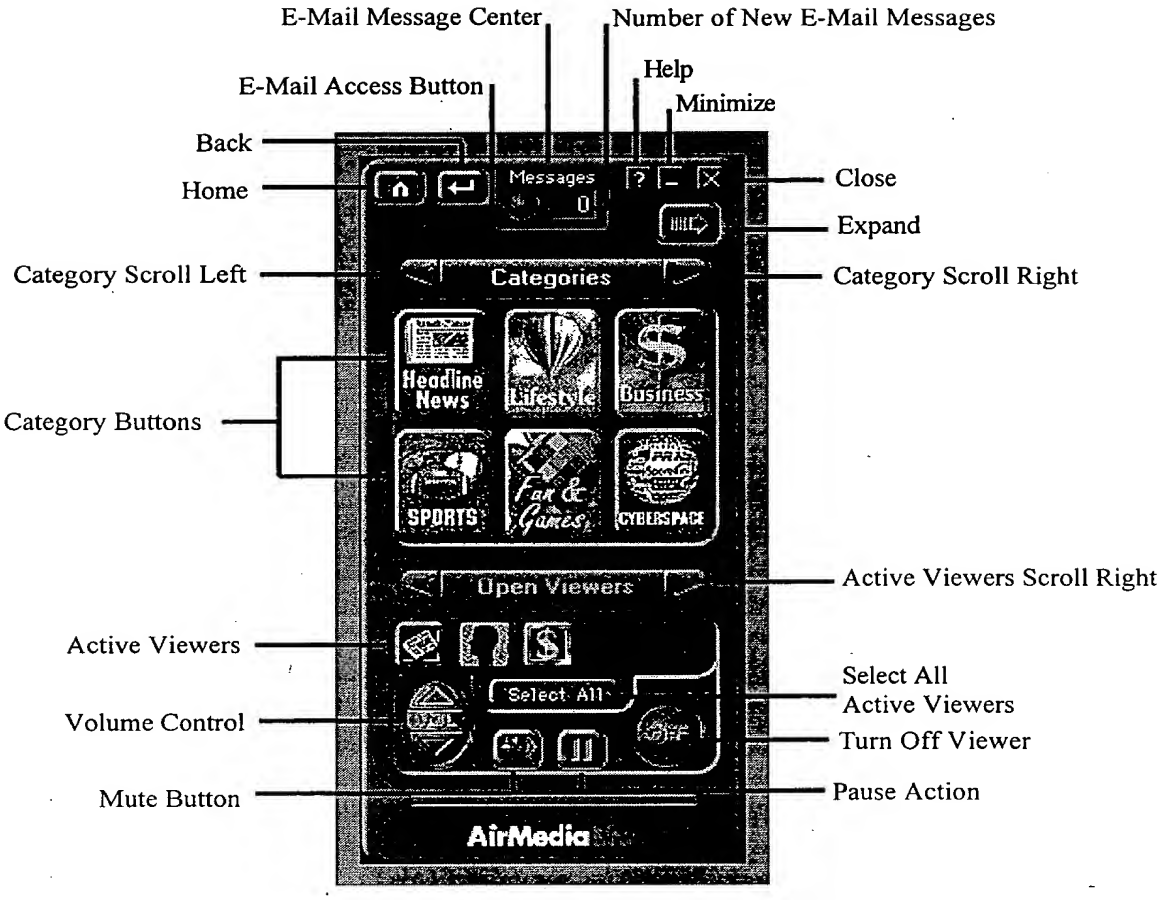

48
Viewers
53
User Preferences DLL
20
Viewer Server
54
Remote Control
55
Error Logging
50
Alert Panel
Database
51
38
Communications Server
Capcode Reprog. and Act. Code Parse DLL
57
AMFlex DLL
46
VxD
44
Receiver Hardware
42
Connection to Database
Connection to Error Logging

FIG. 10


E-Mail Message Center
Number of New E-Mail Messages
E-Mail Access Button
Help
Minimize
Back
Home
Close
Expand
Category Scroll Left
Category Scroll Right
Category Buttons
Active Viewers Scroll Right
Active Viewers
Select All Active Viewers
Volume Control
Turn Off Viewer
Mute Button
Pause Action
Messages
0
Categories
Headline News
Lifestyle
Business
SPORTS
Fun & Games
CYBERSPACE
Open Viewers
Select All
AirMedia

54

FIG. 11

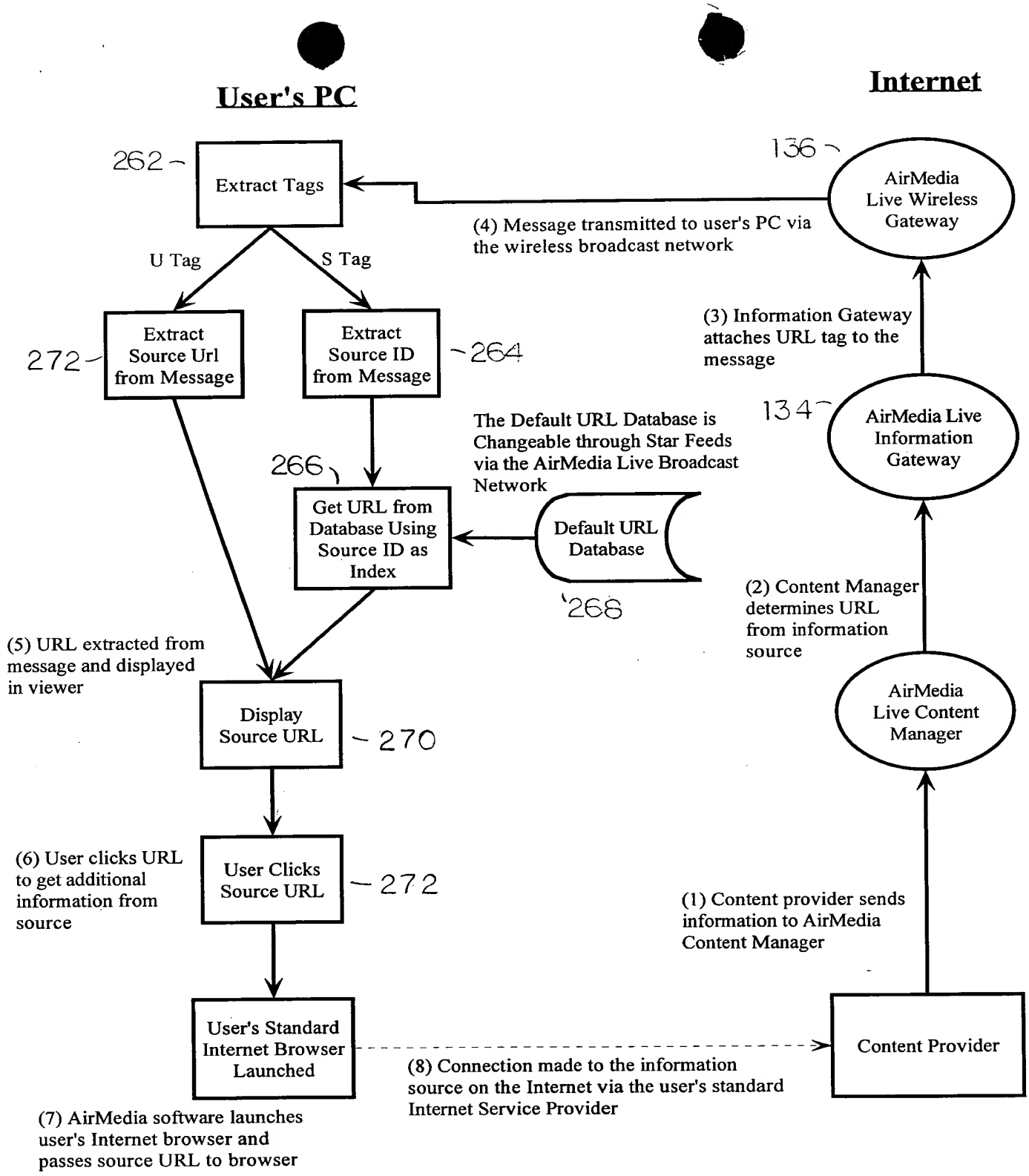

User's PC
Internet
262
Extract Tags
136
AirMedia Live Wireless Gateway
(4) Message transmitted to user's PC via the wireless broadcast network
U Tag
S Tag
(3) Information Gateway attaches URL tag to the message
272
Extract Source Url from Message
Extract Source ID from Message
264
134
AirMedia Live Information Gateway
The Default URL Database is Changeable through Star Feeds via the AirMedia Live Broadcast Network
266
Get URL from Database Using Source ID as Index
Default URL Database
268
(2) Content Manager determines URL from information source
(5) URL extracted from message and displayed in viewer
AirMedia Live Content Manager
Display Source URL
270
(6) User clicks URL to get additional information from source
User Clicks Source URL
272
(1) Content provider sends information to AirMedia Content Manager
User's Standard Internet Browser Launched
Content Provider
(8) Connection made to the information source on the Internet via the user's standard Internet Service Provider
(7) AirMedia software launches user's Internet browser and passes source URL to browser

FIG. 12

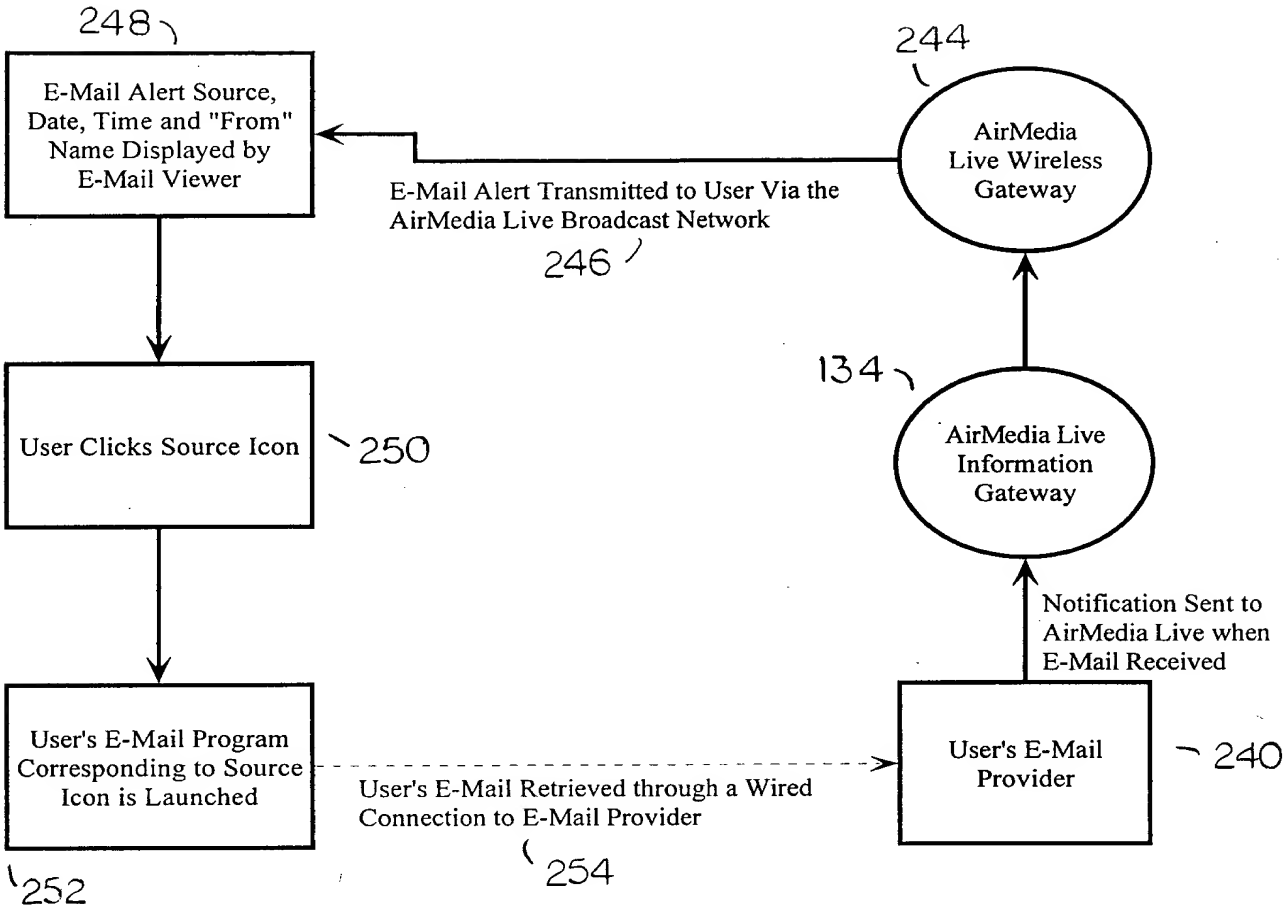

248
E-Mail Alert Source, Date, Time and "From" Name Displayed by E-Mail Viewer
244
AirMedia Live Wireless Gateway
E-Mail Alert Transmitted to User Via the AirMedia Live Broadcast Network
246
User Clicks Source Icon
250
134
AirMedia Live Information Gateway
Notification Sent to AirMedia Live when E-Mail Received
User's E-Mail Program Corresponding to Source Icon is Launched
User's E-Mail Retrieved through a Wired Connection to E-Mail Provider
254
User's E-Mail Provider
240
252

FIG. 13

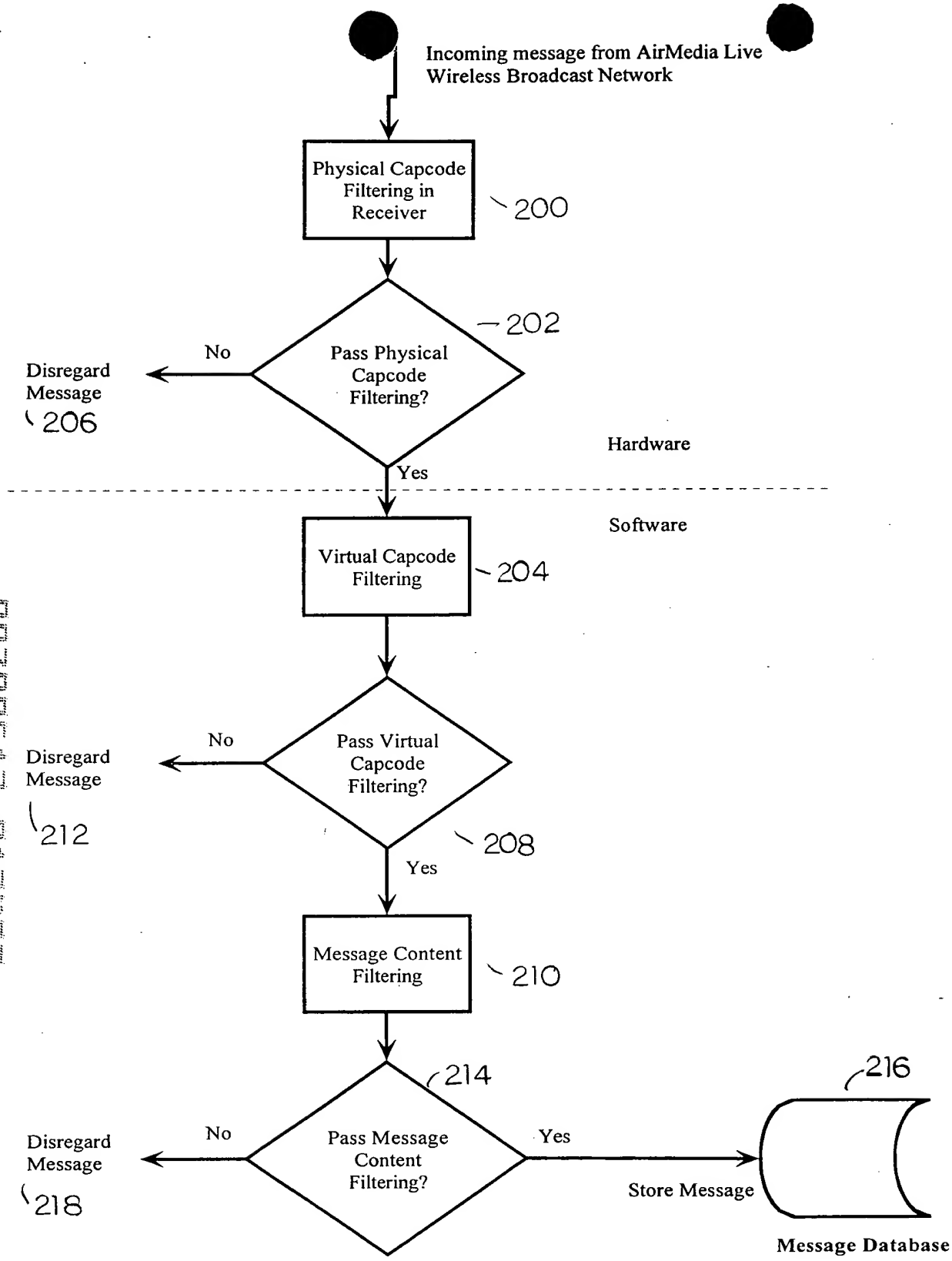

Incoming message from AirMedia Live Wireless Broadcast Network
Physical Capcode Filtering in Receiver
200
Pass Physical Capcode Filtering?
202
No
Disregard Message
206
Hardware
Yes
Software
Virtual Capcode Filtering
204
Pass Virtual Capcode Filtering?
No
Disregard Message
212
208
Yes
Message Content Filtering
210
214
Pass Message Content Filtering?
No
Disregard Message
218
Yes
216
Store Message
Message Database

FIG. 14

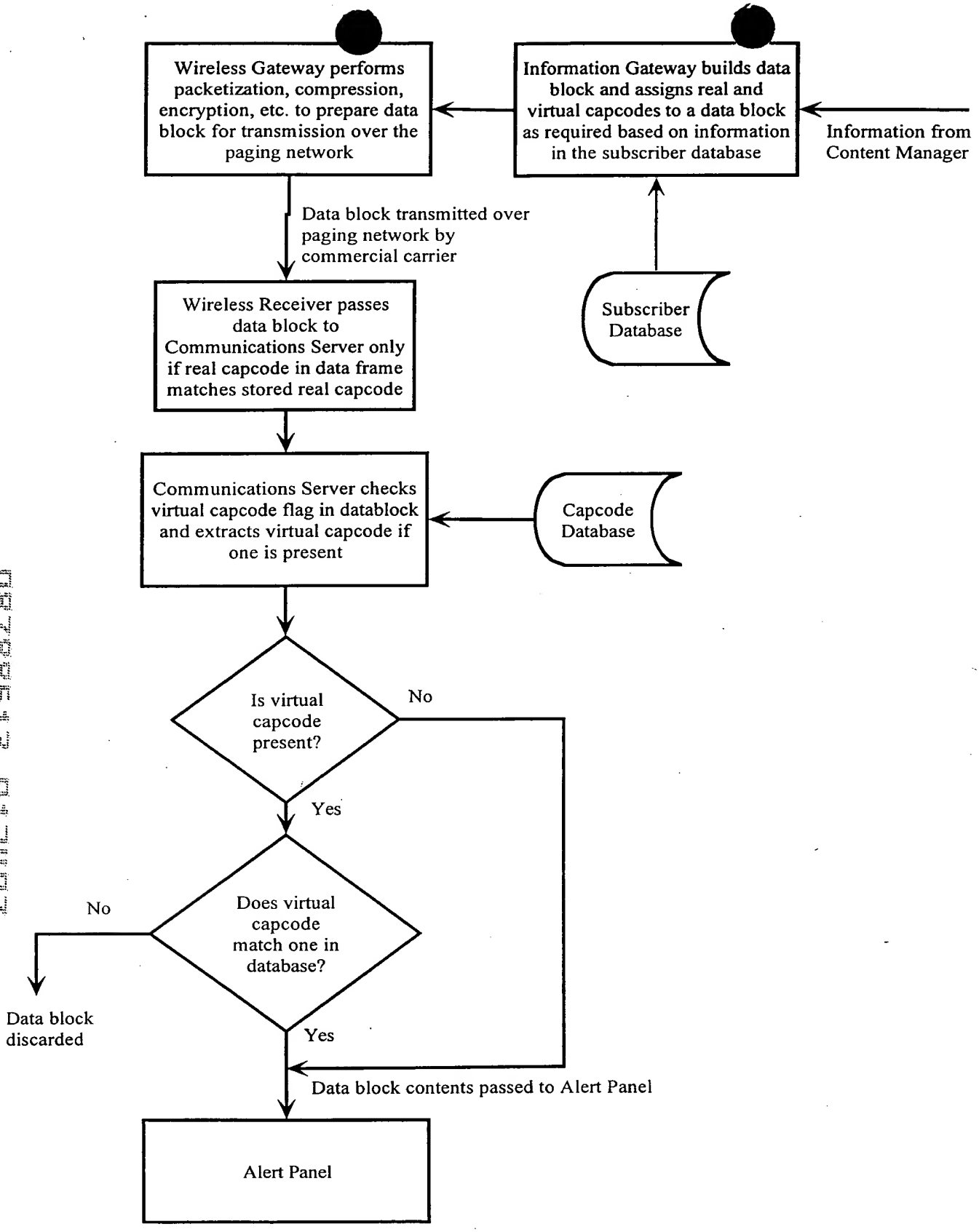

Wireless Gateway performs packetization, compression, encryption, etc. to prepare data block for transmission over the paging network
Information Gateway builds data block and assigns real and virtual capcodes to a data block as required based on information in the subscriber database
Information from Content Manager
Data block transmitted over paging network by commercial carrier
Subscriber Database
Wireless Receiver passes data block to Communications Server only if real capcode in data frame matches stored real capcode
Communications Server checks virtual capcode flag in datablock and extracts virtual capcode if one is present
Capcode Database
Is virtual capcode present?
No
Yes
Does virtual capcode match one in database?
No
Data block discarded
Yes
Data block contents passed to Alert Panel
Alert Panel

FIG. 15

26h210 ET988280

Whole Packets						
	Packet Header	Packet Data				
		column 1	column 2	column 3	....	column s
packet 1:	header 1	x	x	x	....	x
packet 2:	header 2	x	x	x	....	x
.	.	.	.	.	....	.
.	.	.	.	.	....	.
.	.	.	.	.	....	.
packet p:	header p	x	x	x	....	x
packet p + 1	header p + 1	x	x	x	....	x
packet p + 2	header p + 2	x	x	x	....	x
packet p + 3	header p + 3	x	x	x	....	x
.	.	.	.	.	....	.
.	.	.	.	.	....	.
.	.	.	.	.	....	.
packet p + x	header p + x	x	x	x	....	x

Reed Solomon Parity- Check Packets 152

Information Packets 154

FIG. 16

150

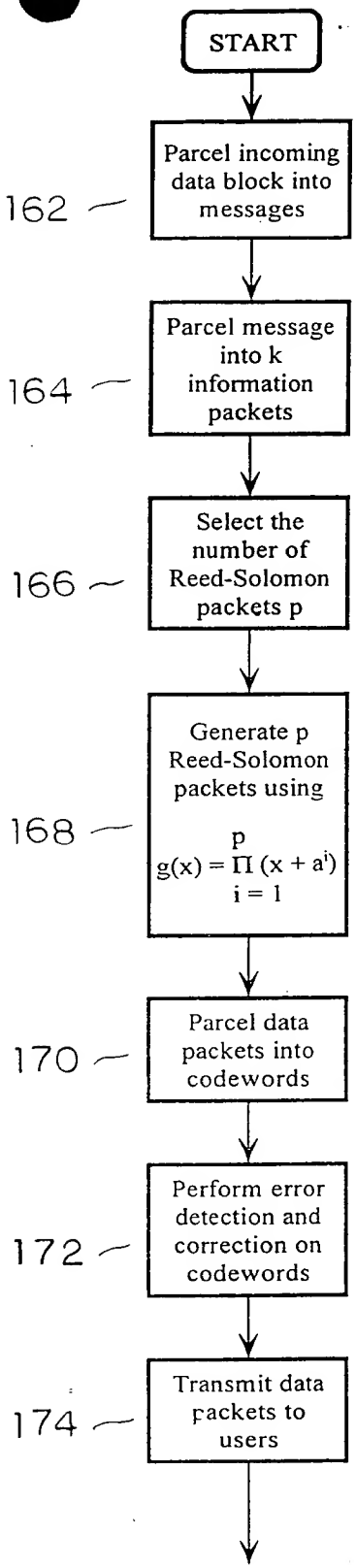

START
Parcel incoming data block into messages
162
Parcel message into k information packets
164
Select the number of Reed-Solomon packets p
166
Generate p Reed-Solomon packets using
$g(x) = \prod_{i=1}^{p} (x + a^i)$
168
Parcel data packets into codewords
170
Perform error detection and correction on codewords
172
Transmit data packets to users
174

FIG. 17

160

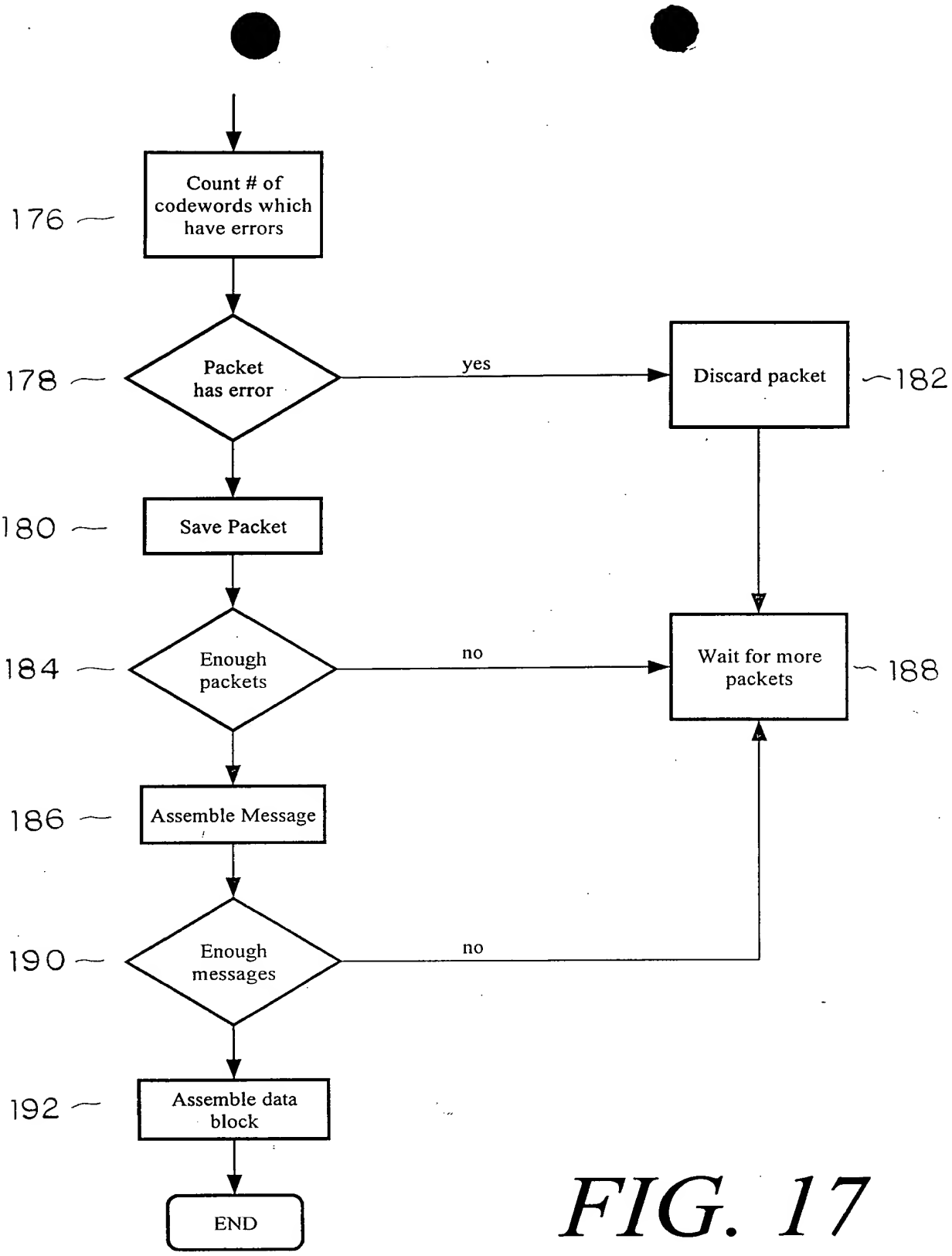
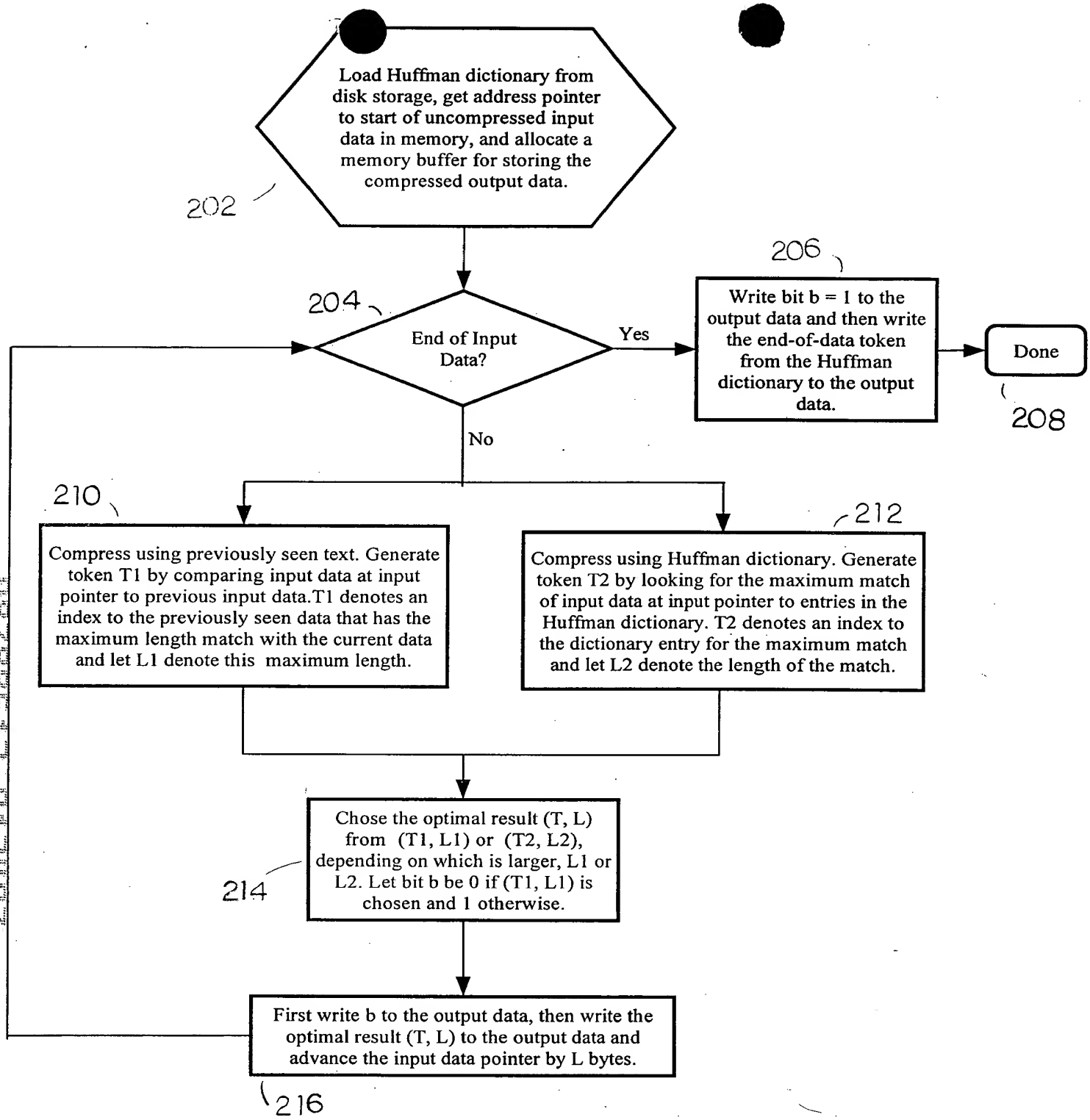
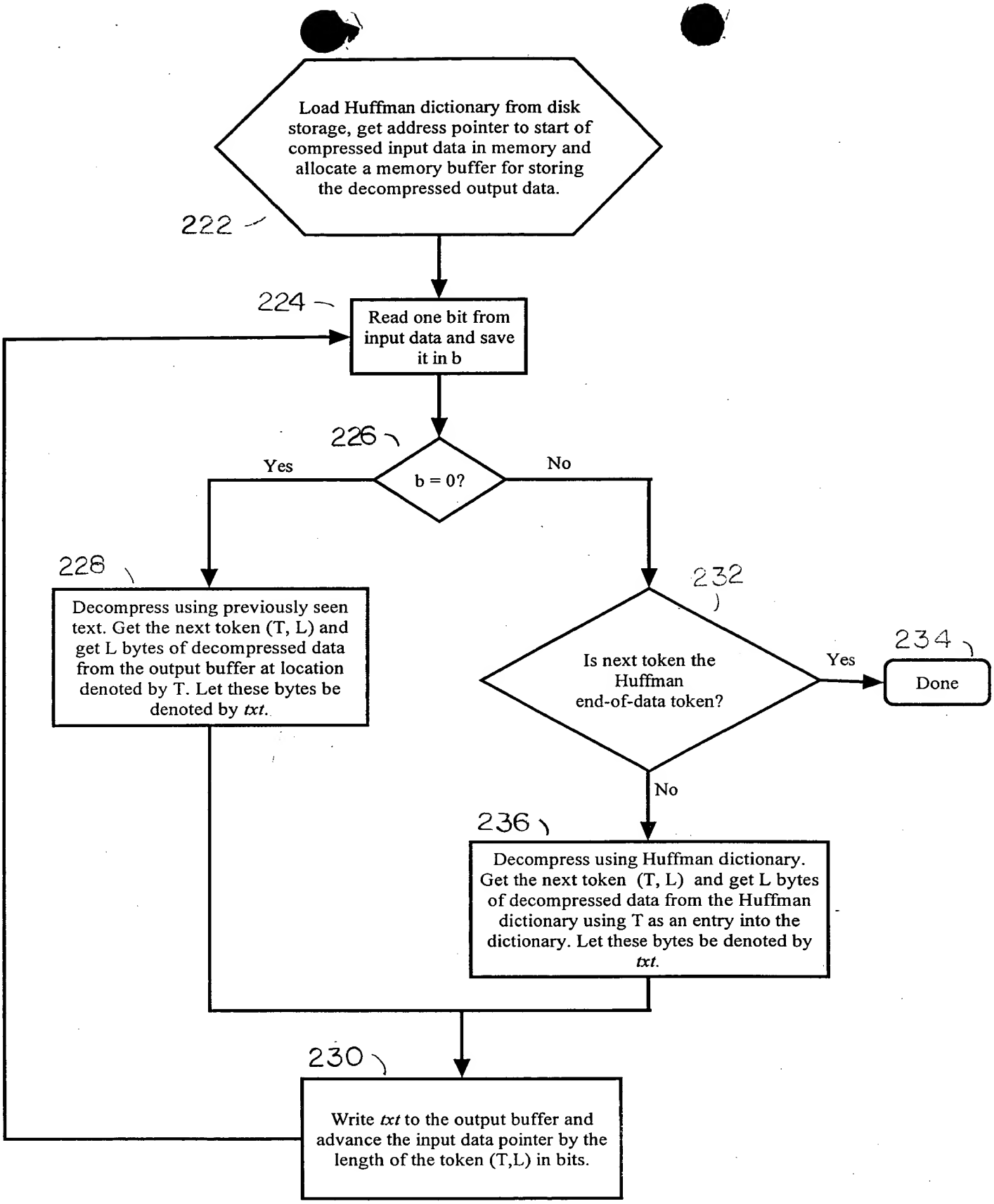

Count # of codewords which have errors
176
Packet has error
178
yes
Discard packet
182
Save Packet
180
Enough packets
184
no
Wait for more packets
188
Assemble Message
186
Enough messages
190
no
Assemble data block
192
END

FIG. 17 (cont.)


Load Huffman dictionary from disk storage, get address pointer to start of uncompressed input data in memory, and allocate a memory buffer for storing the compressed output data.
202
End of Input Data?
204
Yes
No
Write bit b = 1 to the output data and then write the end-of-data token from the Huffman dictionary to the output data.
206
Done
208
Compress using previously seen text. Generate token T1 by comparing input data at input pointer to previous input data.T1 denotes an index to the previously seen data that has the maximum length match with the current data and let L1 denote this maximum length.
210
Compress using Huffman dictionary. Generate token T2 by looking for the maximum match of input data at input pointer to entries in the Huffman dictionary. T2 denotes an index to the dictionary entry for the maximum match and let L2 denote the length of the match.
212
Chose the optimal result (T, L) from (T1, L1) or (T2, L2), depending on which is larger, L1 or L2. Let bit b be 0 if (T1, L1) is chosen and 1 otherwise.
214
First write b to the output data, then write the optimal result (T, L) to the output data and advance the input data pointer by L bytes.
216

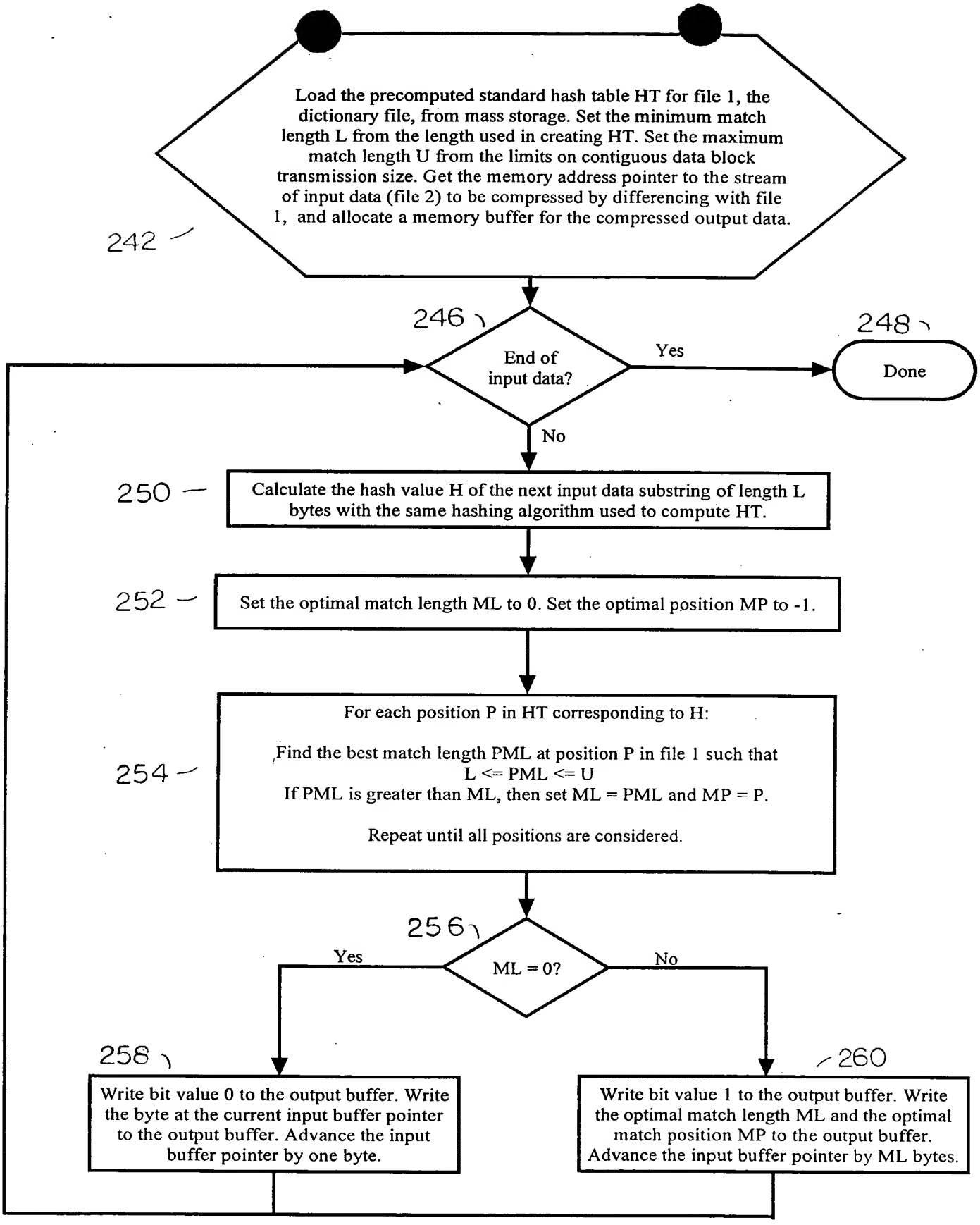
200

FIG. 18(a)


Load Huffman dictionary from disk storage, get address pointer to start of compressed input data in memory and allocate a memory buffer for storing the decompressed output data.
222
224
Read one bit from input data and save it in b
226
b = 0?
Yes
No
228
Decompress using previously seen text. Get the next token (T, L) and get L bytes of decompressed data from the output buffer at location denoted by T. Let these bytes be denoted by txt.
232
Is next token the Huffman end-of-data token?
Yes
234
Done
No
236
Decompress using Huffman dictionary. Get the next token (T, L) and get L bytes of decompressed data from the Huffman dictionary using T as an entry into the dictionary. Let these bytes be denoted by txt.
230
Write txt to the output buffer and advance the input data pointer by the length of the token (T,L) in bits.

220

FIG. 18(b)


Load the precomputed standard hash table HT for file 1, the dictionary file, from mass storage. Set the minimum match length L from the length used in creating HT. Set the maximum match length U from the limits on contiguous data block transmission size. Get the memory address pointer to the stream of input data (file 2) to be compressed by differencing with file 1, and allocate a memory buffer for the compressed output data.
242
246
End of input data?
Yes
248
Done
No
250
Calculate the hash value H of the next input data substring of length L bytes with the same hashing algorithm used to compute HT.
252
Set the optimal match length ML to 0. Set the optimal position MP to -1.
254
For each position P in HT corresponding to H:
Find the best match length PML at position P in file 1 such that L <= PML <= U
If PML is greater than ML, then set ML = PML and MP = P.
Repeat until all positions are considered.
256
ML = 0?
Yes
No
258
Write bit value 0 to the output buffer. Write the byte at the current input buffer pointer to the output buffer. Advance the input buffer pointer by one byte.
260
Write bit value 1 to the output buffer. Write the optimal match length ML and the optimal match position MP to the output buffer. Advance the input buffer pointer by ML bytes.

240

FIG. 19(a)

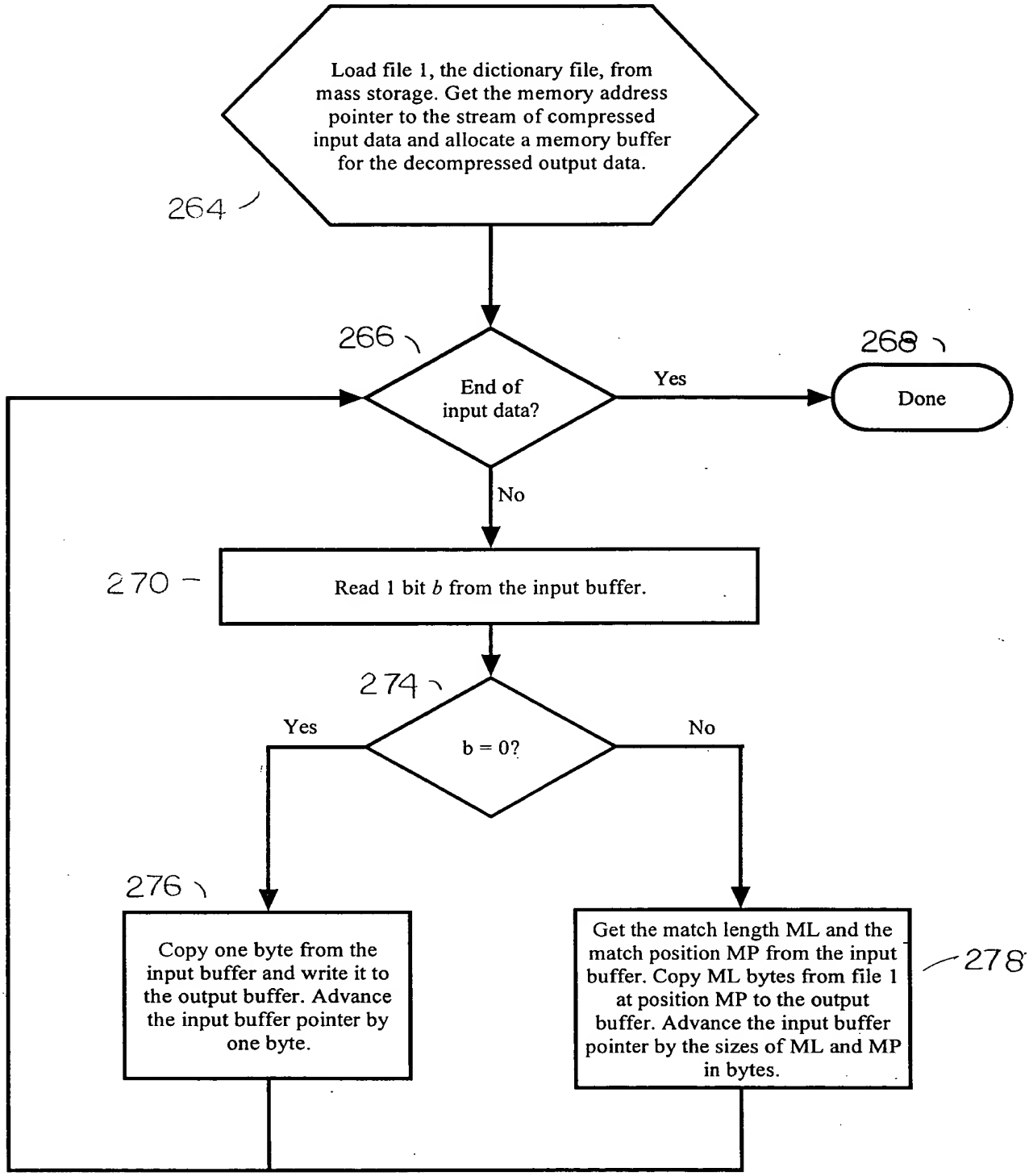

Load file 1, the dictionary file, from mass storage. Get the memory address pointer to the stream of compressed input data and allocate a memory buffer for the decompressed output data.
264
266
End of input data?
Yes
268
Done
No
270
Read 1 bit b from the input buffer.
274
b = 0?
Yes
No
276
Copy one byte from the input buffer and write it to the output buffer. Advance the input buffer pointer by one byte.
Get the match length ML and the match position MP from the input buffer. Copy ML bytes from file 1 at position MP to the output buffer. Advance the input buffer pointer by the sizes of ML and MP in bytes.
278

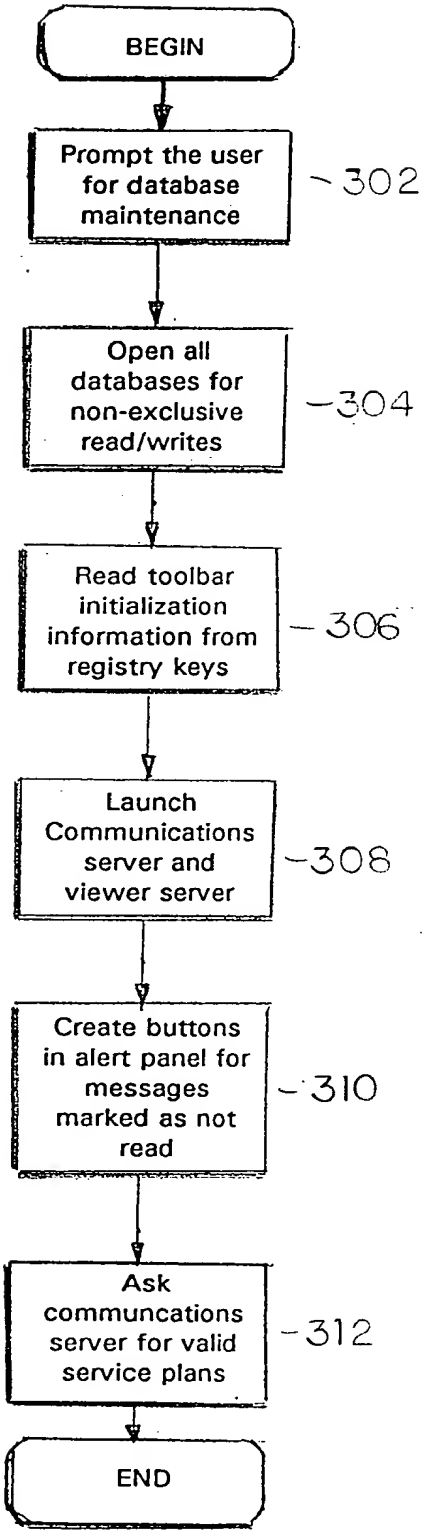
FIG. 19(b)

262

50

FIG. 20

Case 1:07-cv-00467-SLR Document 1-2 Filed 07/26/2007


BEGIN
Prompt the user for database maintenance
302
Open all databases for non-exclusive read/writes
304
Read toolbar initialization information from registry keys
306
Launch Communications server and viewer server
308
Create buttons in alert panel for messages marked as not read
310
Ask communcations server for valid service plans
312
END

300

FIG. 21

Z6h2T0" ET98Z80

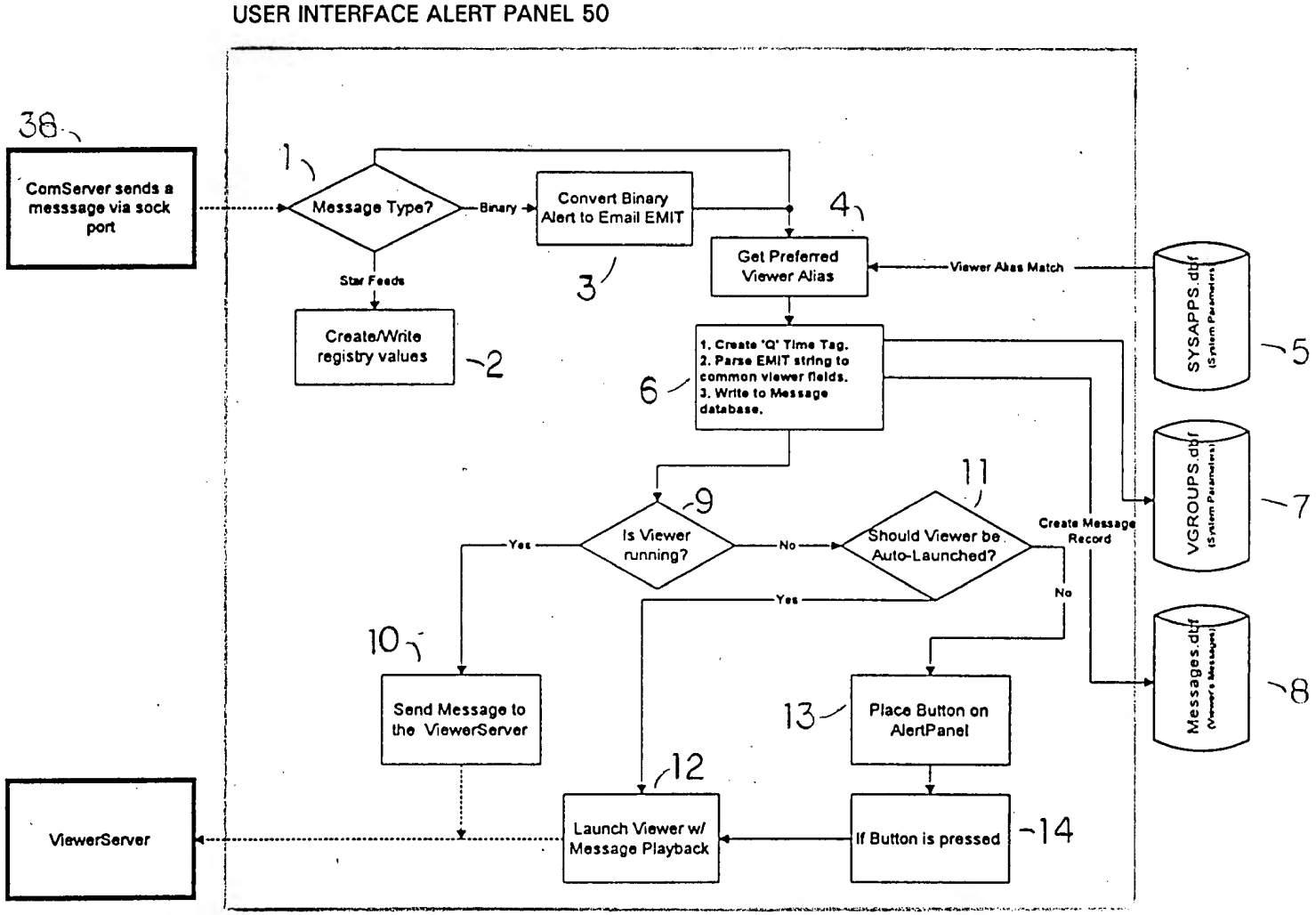


FIG.22

26h210 ET9887B0

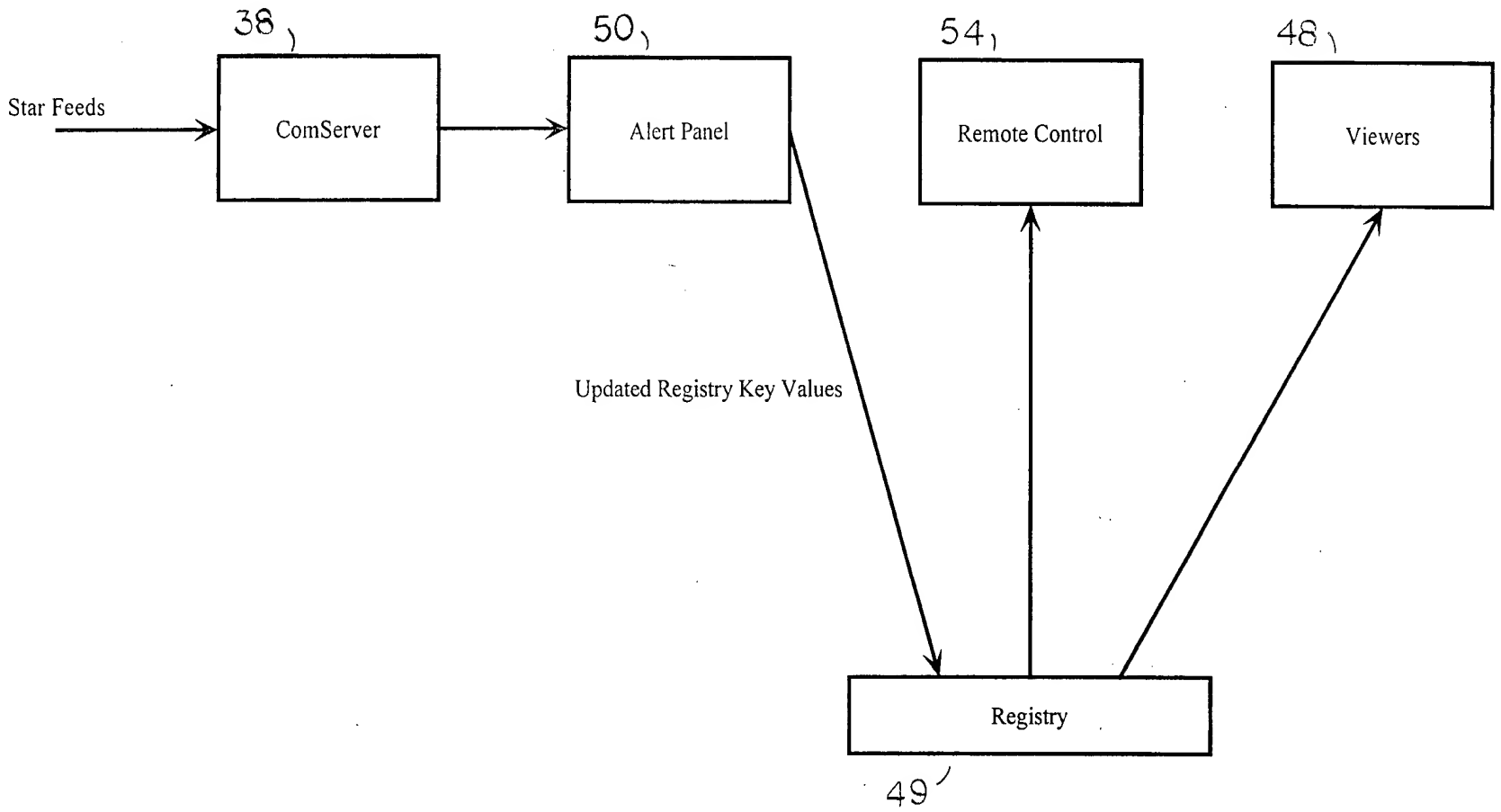


FIG. 23

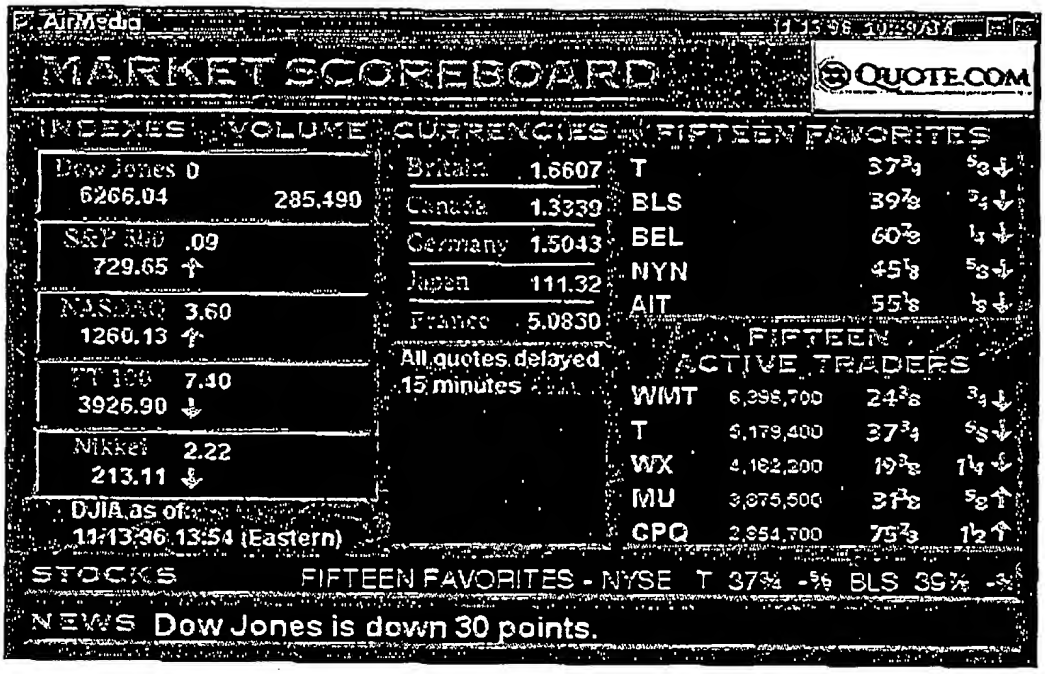


FIG. 24(a)

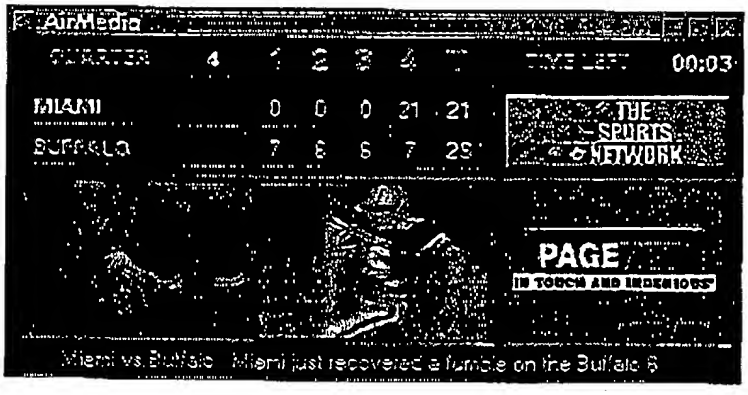


FIG. 24(b)

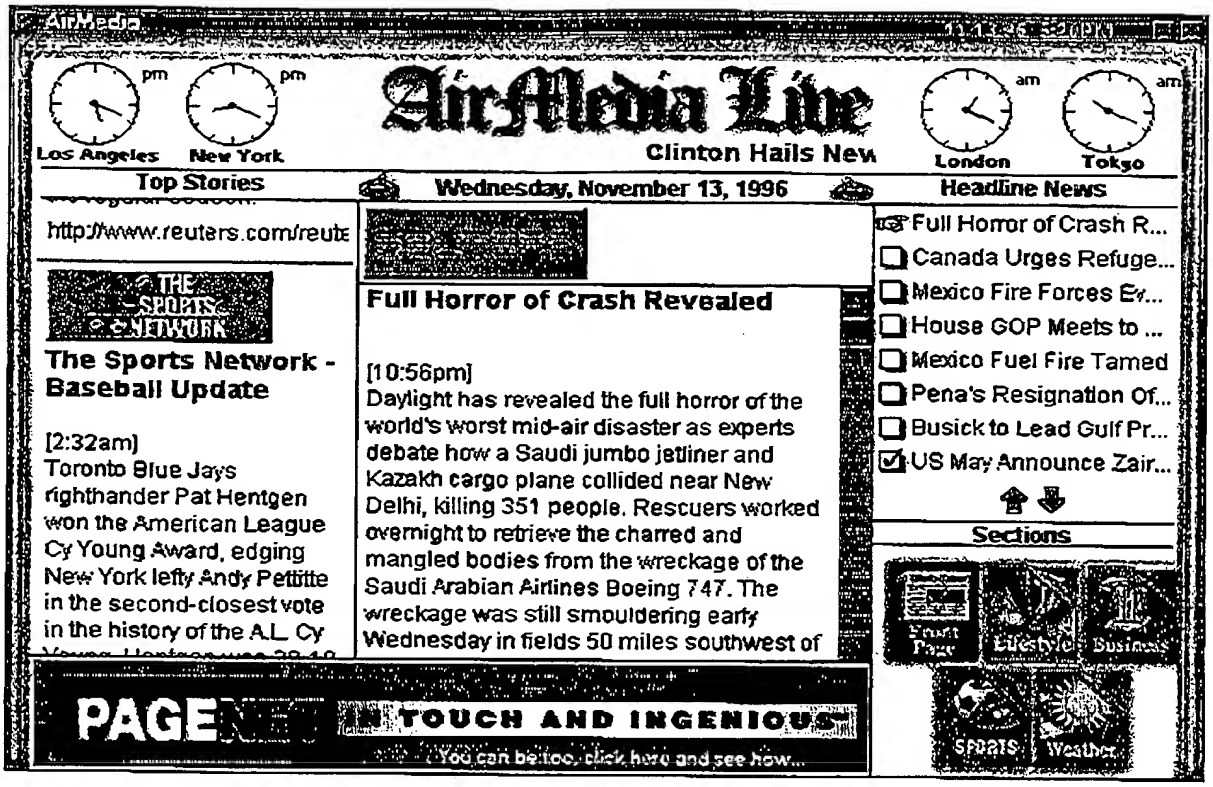


FIG. 24(c)

FIG. 24(d)

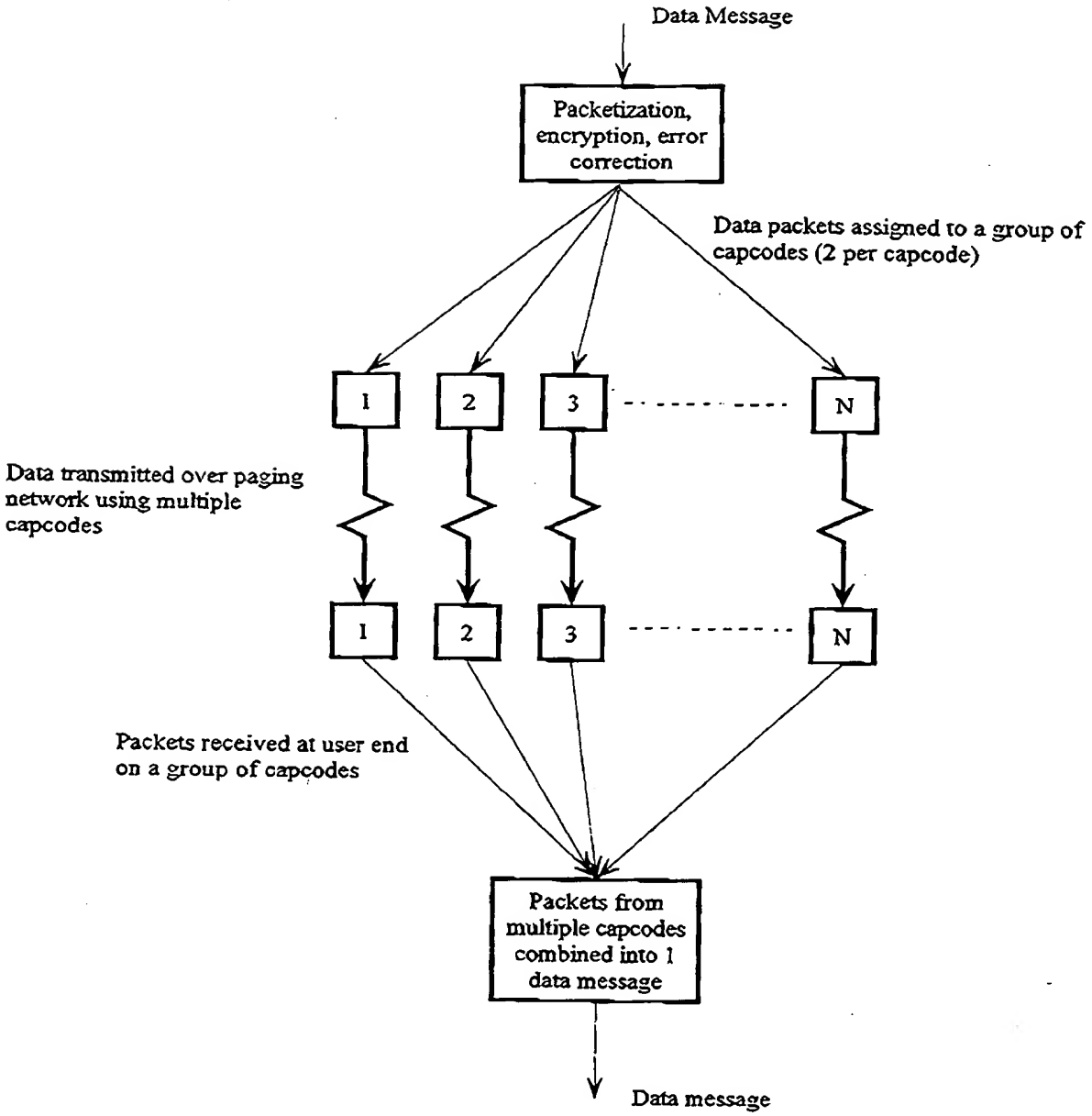

Data Message
Packetization, encryption, error correction
Data packets assigned to a group of capcodes (2 per capcode)
1
2
3
N
Data transmitted over paging network using multiple capcodes
1
2
3
N
Packets received at user end on a group of capcodes
Packets from multiple capcodes combined into 1 data message
Data message

FIG. 25